

DSIB01 Autumn 2021

01 Tooling



Mgr. Eliška Chalupová
375973@mail.muni.cz

Practicals overview

- Linux set-up
 - Version Control Systems
 - Git
 - Markdown
 - Coding Environment
-
- Hands-on example
 - Project set-up

Linux set-up

VirtualBox



1. Get VirtualBox for Windows at <https://www.virtualbox.org/wiki/Downloads>
2. Download a pre-installed machine (DSIB_VM.ova) from <https://drive.google.com/file/d/1DQYTS6YLkomvgGglu1ce5LAcLCayx9AI/view?usp=sharing>
3. Import the machine
4. Start & Log-in (usr: student, pswd: 1234)



Version Control System

Benefits

1. A complete long-term change history of every file

Version Control System Benefits

1. A complete long-term change history of every file
2. Enables recovery



Version Control System Benefits

1. A complete long-term change history
2. Enables recovery
3. Simplifies team work

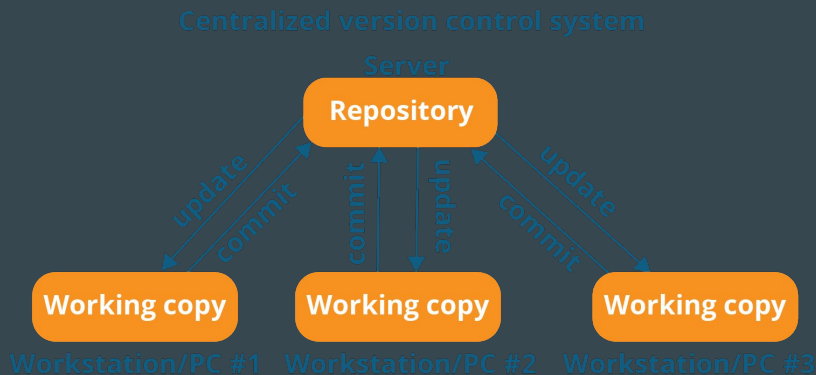


Version Control System Types

1. Local VCS
2. Centralized VCS
3. Distributed VCS

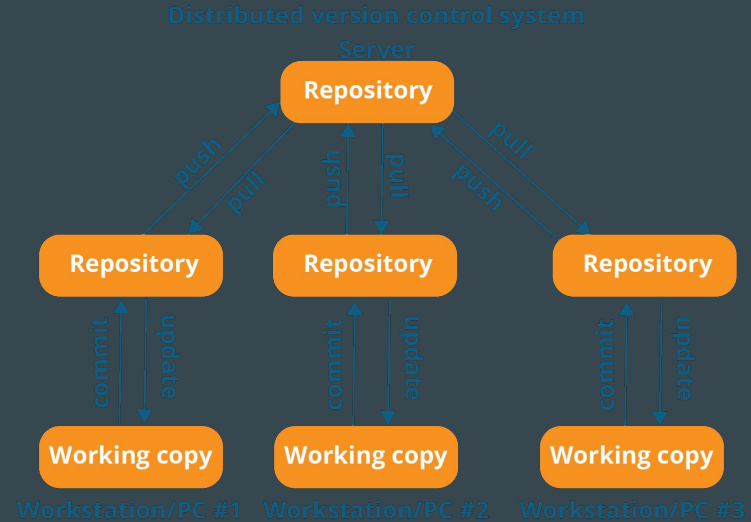
Version Control System Types

1. Local VCS
2. Centralized VCS
3. Distributed VCS



Version Control System Types

1. Local VCS
2. Centralized VCS
3. **Distributed VCS**



Git

Repository

Anyone with a copy of the repo can:

- Check the history
- Clone
- Commit
- Branch
- Merge
- Compare changes
- ...

Working models:

- Fork and Pull
- Shared Repository

Git
Hosting



GitLab

ATLASSIAN



Bitbucket



ProjectLocker



beanstalk



HelixTeamHub



Codeberg

codebase
by KRYSTAL



Git Hosting



<https://education.github.com/pack>

Hands-on

1. Set-up Git

- Check Git availability on the machine, and if needed download Git from <https://git-scm.com/downloads>
- Choose a Git hosting page & set-up an account
- Fork the Demo repository at <https://github.com/eliska-chalupova/DSIB01demo> or create a repository with the same structure





Git Windows



Git Bash

<https://gitforwindows.org/>

Git GUI Clients

Client	Platform	Website
 GitKraken	Linux, Mac, Windows	https://www.gitkraken.com/
 SmartGit	Linux, Mac, Windows	https://www.syntevo.com/smartgit/
 GitHub Desktop	Mac, Windows	https://desktop.github.com/
 Sourcetree	Mac, Windows	https://www.sourcetreeapp.com/

Git GUI Clients

The screenshot displays a Git GUI client interface with the following components:

- Repository:** support.gitkraken.com
- Branch:** v7-5-docs-update
- Graph:** A vertical commit history graph showing the current branch (v7-5-docs-update) and its relationship to other branches (production, dev) and tags (v1.9.0).
- Commit List:** A list of commits with their messages and timestamps. The most recent commit is // WIP (1 hour ago). Other recent commits include "Additional edits", "Purchase FAQ change", "v7.5 Trial page changes", and "Merge branch 'dev' into production".
- Local Branches:** dev, production, v7-5-docs-update (checked).
- Remote Branches:** Axosoft, Axozilla, MannyGrill, diano28, jjsilva4, jkelroy.
- Unstaged Files (5):** docs/account/faq.md, docs/account/trial-activation.md, docs/img/documentation/gett.../commit-panel.png, docs/img/documentation.../commit-panel@2x.png, docs/integrations/trello.md.
- Staged Files (0):** No files are currently staged.
- Commit Message:** A text input field for the current commit message, with an "Amend" button.
- Summary/Description:** A section for providing a summary or description of the commit.

Hands-on

2. GUI Client

- Download and run a GUI client of Your choice
- Open Your repository
- Look around

Git

Good practices

- Create branches
- Make small commits
- Write meaningful commit messages
- Use pull requests
- Review and discuss code
- Rebase often

Markdown Briefly

“Markdown is a lightweight markup language for creating formatted text using a plain-text editor.”

MARKDOWN SYNTAX

Markdown is a way to style text on the web. You control the display of the document; formatting words as bold or italic, adding images, and creating lists are just a few of the things we can do with Markdown. Mostly, Markdown is just regular text with a few non-alphabetic characters thrown in, like # or *.



HEADERS

```
# This is an <h1> tag
## This is an <h2> tag
#### This is an <h4> tag
```

EMPHASIS

```
*This text will be italic*
_This will also be italic_

**This text will be bold**
__This will also be bold__

*You **can** combine them*
```

LISTS

Unordered

- Item 1
- Item 2
- Item 2a
- Item 2b

Ordered

1. Item 1
 2. Item 2
 3. Item 3
- Item 3a
 - Item 3b

IMAGES

```
![GitHub Logo](/images/logo.png)
```

Format: ![Alt Text](url)

LINKS

```
http://github.com - automatic!
```

```
[GitHub](http://github.com)
```

BLOCKQUOTES

As Kanye West said:

```
> We're Living the future so
> the present is our past.
```

As Kanye West said:

```
| We're living the future so
| the present is our past.
```

BACKSLASH ESCAPES

Markdown allows you to use backslash escapes to generate literal characters which would otherwise have special meaning in Markdown's formatting syntax.

```
\*literal asterisks\*
```

```
"literal asterisks"
```

Markdown provides backslash escapes for the following characters:

\	backslash	()	parentheses
#	hash mark	#	hash mark
*	asterisk	+	plus sign
_	underscore	-	minus sign (hyphen)
	curly braces	.	dot
[]	square brackets	!	exclamation mark

GITHUB FLAVORED MARKDOWN

GitHub.com uses its own version of the Markdown syntax that provides an additional set of useful features, many of which make it easier to work with content on GitHub.com.



USERNAME @MENTIONS

Typing an @ symbol, followed by a username, will notify that person to come and view the comment. This is called an "mention", because you're mentioning the individual. You can also @mention teams within an organization.

ISSUE REFERENCES

Any number that refers to an Issue or Pull Request will be automatically converted into a link.

```
#1
default#1
default/github-flavored-markdown#1
```

TASK LISTS

- [x] @mentions, #refs, [links](), **formatting**, and tags supported
- [x] list syntax required (any unordered or ordered list supported)
- [x] this is a complete item
- [] this is an incomplete item

- ☑ @mentions, #refs, links, formatting, and tags supported
- ☑ list syntax required (any unordered or ordered list supported)
- ☑ this is a complete item
- ☐ this is an incomplete item

FENCED CODE BLOCKS

Markdown converts text with four leading spaces into a code block; with GFM you can wrap your code with ````` to create a code block without the leading spaces. Add an optional language identifier and your code with get syntax highlighting.

```
```javascript
function test() {
 console.log("look ma, no spaces");
}
```
```

```
function test() {
  console.log("look ma, no spaces");
}
```

TABLES

You can create tables by assembling a list of words and dividing them with hyphens - (for the first row), and then separating each column with a pipe | :

```
First Header	Second Header
Content cell 1 | Content cell 2
Content: column 1 | Content: column 2
```

| First Header | Second Header |
|------------------|------------------|
| Content cell 1 | Content cell 2 |
| Content column 1 | Content column 2 |

EMOJI

To see a list of every image we support, check out www.emoji-cheat-sheet.com

```
GitHub supports emoji!
:+1: :sparkles: :camel: :tada:
:rocket: :metal: :octocat:
```

GitHub supports emoji!

Hands-on

3. Markdown

- Write a text in Your Project Report / README file
- Use elements of the Markdown syntax

Coding Environment

Code Editor vs IDE



IDE vs Code Editor

Comparison Chart

| IDE | Code Editor |
|--|--|
| An IDE is a set of software development tools designed to make coding easier. | Code editor is a developer's tool designed to edit the source code of computer programs. |
| It consolidates many of the functions like code creation, building and testing, together in a single framework service or application. | It is a text editor with powerful built-in features and specialized functionalities to simplify and accelerate code editing process. |
| Key features include text editing, compiling, debugging, GUI, syntax highlighting, unit testing, code completion, and more. | Key features include syntax highlighting, printing, multiview, and preview window. |
| Some popular IDEs are Eclipse, IntelliJ IDEA, Visual Studio, NetBeans, etc. | Some common code editors include Atom, Sublime Text, Brackets, Visual Studio Code, etc. |



Sublime Text



Visual Studio Code



Brackets







Coding Environment

Free Code Editors

| Code Editor | Platform | Website |
|---|---------------------|---|
|  | Linux, Mac, Windows | https://atom.io/ |
| 
Visual Studio Code | Linux, Mac, Windows | https://code.visualstudio.com/ |
|  | Windows | https://notepad-plus-plus.org/ |
|  | Linux | https://www.vim.org/ |

Coding Environment

Free IDEs

| IDE | Platform | Website |
|--|---------------------|---|
|  IntelliJ IDEA | Linux, Mac, Windows | https://www.jetbrains.com/shop/eform/students |
|  Visual Studio | Mac, Windows | https://visualstudio.microsoft.com/vs/community/ |
|  Apache NetBeans IDE | Linux, Mac, Windows | https://netbeans.apache.org/ |
|  eclipse | Linux | https://www.eclipse.org/eclipseide/ |

Coding Environment Jupyter Notebook

“The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text.”

The image displays two overlapping Jupyter Notebook windows. The background window shows the 'Welcome to the Jupyter Notebook Server' page, which includes a warning box stating 'Don't rely on this server' and instructions on how to run Python code in a cell. The foreground window is titled 'Exploring the Lorenz System' and contains the following content:

Exploring the Lorenz System

In this Notebook we explore the [Lorenz system](#) of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters (σ, β, ρ) are varied, including what are known as *chaotic solutions*. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.

In [7]: `interact(Lorenz, N=fixed(10), angle=(0., 360.),
sigma=(0.0, 50.0), beta=(0., 5), rho=(0.0, 50.0))`

The interactive plot shows five sliders for parameters: angle (308.2), max_time (12), σ (10), β (2.6), and ρ (28). Below the sliders is a 3D visualization of the Lorenz attractor, a complex, butterfly-shaped trajectory in phase space.

Coding Environment

Jupyter Notebook

Uses include:

- data cleaning and transformation,
- numerical simulation,
- statistical modeling,
- data visualization,
- machine learning, and much more.

The image displays two overlapping Jupyter Notebook windows. The background window shows the 'Welcome to the Jupyter Notebook Server' page, which includes a warning box stating 'Don't rely on this server' and instructions on how to run code. The foreground window is titled 'Lorenz Differential Equations (autosaved)' and contains the following content:

Exploring the Lorenz System

In this Notebook we explore the [Lorenz system](#) of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters (σ, β, ρ) are varied, including what are known as *chaotic solutions*. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.

```
In [7]: interact(Lorenz, N=fixed(10), angle=(0., 360.),
                sigma=(0.0, 50.0), beta=(0., 5), rho=(0.0, 50.0))
```

The interactive interface shows sliders for the following parameters:

- angle: 308.2
- max_time: 12
- σ : 10
- β : 2.6
- ρ : 28

Below the sliders is a 3D plot of the Lorenz attractor, showing its characteristic butterfly shape with multiple colored trajectories.

Coding Environment Jupyter Notebook

<https://jupyter.org/try>

The image displays two overlapping Jupyter Notebook windows. The background window shows the 'Welcome to the Jupyter Notebook' page, which includes a warning box stating 'Don't rely on this server' and instructions on how to run code cells. The foreground window is titled 'Exploring the Lorenz System' and contains the following content:

Exploring the Lorenz System

In this Notebook we explore the [Lorenz system](#) of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters (σ, β, ρ) are varied, including what are known as *chaotic solutions*. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.

```
In [7]: interact(Lorenz, N=fixed(10), angle=(0., 360.),
                sigma=(0.0, 50.0), beta=(0., 5), rho=(0.0, 50.0))
```

The code cell is followed by an interactive interface with sliders for the following parameters:

- angle: 308.2
- max_time: 12
- σ : 10
- β : 2.6
- ρ : 28

Below the sliders is a 3D plot of the Lorenz attractor, showing its characteristic butterfly shape with multiple colored trajectories.

Hands-on

4. Environment

- Get a Code Editor or IDE of Your choice
- Open the repository / selected file
- Make changes in one of the files in the /demo folder
- Push the changes to Your remote repository

Hands-on

5. Play with Git

Try out diverse Git commands:

- Create a local / remote branch
- Commit
- Check status
- Create stash
- Push
- Merge two branches
- Add a tag
- ...

Project Set-up

- Set up an account on a Git-hosting site
- Create a repository with the structure as in <https://github.com/eliska-chalupova/DSIB01demo>
- The repository will be used for all the materials for Your project throughout the course
- Use the README file to write Your project report
- Write & discuss all the results throughout the course in the project report
- You will present Your project at the colloquium by the end of semester

Questions ?



Mgr. Eliška Chalupová
375973@mail.muni.cz